

A
Project Report
on
**IDENTIFYING OF FAKE PROFILES ACROSS ONLINE USING NEURAL
NETWORKS**

Submitted for partial fulfilment of the requirements for the award of the degree of
BACHELOR OF ENGINEERING
in

COMPUTER SCIENCE AND ENGINEERING
By

J Prathima (18N81A0509)
Y Pallavi (18N1A0512)
G Nishika (18N81A0518)
K Meghana (18N1A0530)

Under the guidance of
Mr. A. VENU GOPAL
Assistant Professor
Department of CSE



B. Tech * MBA
SPHOORTHY ENGINEERING COLLEGE
Department of Computer Science and Engineering
(Affiliated to JNTUH & Recognized by AICTE)
Nadergul, Saroornagar Mandal, Hyderabad – 501 510
Academic Year: 2021-22



B. Tech * MBA

CERTIFICATE

*This is to certify that this Project Seminar Report entitled “**IDENTIFYING OF FAKE PROFILES ACROSS ONLINE USING NEURAL NETWORKS** ” is a bonafide work carried out by **J Prathima (18N81A0509), Y Pallavi (18N81A0509), G Nishika (18N81A0518), K Meghana(18N81A0530)** in partial fulfillment of the requirements for the award of degree of **Bachelor of Engineering in Computer Science And Engineering** from **Sphoorthy Engineering College**, affiliated to **Jawaharlal Nehru Technological University Hyderabad, Hyderabad**, during the Academic Year 2021-22 under our guidance and supervision.*

The results embodied in this report have not been submitted to any other university or institute for the award of any degree or diploma.

Internal Guide

Mr. A. Venu Gopal
Assistant Professor
Department of CSE
SPHN.

Head of the Department

P. Ram Mohan Rao
Head
Department of CSE
SPHN.

IDENTIFYING OF FAKE PROFILES USING NEURAL NETWORKS

Abstract:

In seeing the present condition, online social networks are engaging with the majority of the people. From child to adult, all are spending a considerable time on these platforms either by exchanging information or making efficient communication with others. But nowadays, these social networking sites are suffering from a lot of fake accounts in taking advantage of vulnerabilities, either taking the benefits or targeting accounts attempting cybercrime

1.INTRODUCTION

1.1 Research Problem

The concern about fake profile is protecting personal data or information from cyber attacks known as phishing attacks. The cyber attackers are often use this in stealing of information. In detecting of passwords, sharing of irrelevant contents, raising awareness this type of profiles are involved in all unlawful activity. In managing and taking the advantages of the critical situation this can be lead to the anonymity through a longer way. For reducing the incidents like trolling, hacking and cyber bullying this is need to be identified.

1.2 Research Rationale

In securing the all types of social accounts and keeping the users away from the cyber hackers this is necessary to identify those and using of ANNs model this can improved in more better way.

1.3 Research Approach

In regards to this, an "artificial neural network" system has been introduced as a part of the computer system. It is designed for simulating in a way in which the human brain possesses and analyses information. The inductive research approach can be considered for this type. In viewing the existing process and situations this can be observed through the patterns and system regularities. In taking the technical advantage ANN model need to be used effectively. It can be described as a foundation of artificial intelligence which will solve the problem in proving the difficulty according to human standards. Therefore "artificial neural networks" (ANNs) are introduced as a process of modeling, allowing the human nervous system through learning technique. By depending on the prediction, this detection process is revealing about the "user-level activities' ". User influence is also vital in reporting about the abnormalities. The social influence upon users can be assessed with the two types of factors. One is to find the user's impact upon others, and the other is to give the user importance. The evaluation is also based on the "fine-grained feature'.

1.4 Aim and Objectives

The main aim is defined in reducing of fake accounts in social network by involving of ANN process. The objectives are –

- To identify of fake accounts in social networks
- To apply the ANN's through machine learning techniques.



Figure1: Social Network

2. Literature Survey

2.1 Survey of Major Area relevant to Project

In viewing Ramalingam and Chinnaiah (2017), most social networking sites cannot notify the fake profiles. Therefore, the discrimination can be seen in between the fake and real profiles, which is technically challenging for most of the users. The existing model is used for this related research study. From the available dataset, each of the social sites is extracting features by using the component analysis. Apart from this, the "Sybil frame" can be used as a multi-level mechanism which is detecting the Sybil's of Twitter and Facebook (Hajdu *et al.* 2019). There are also two types of approaches available for this type: structure-based and content-based. The "vote trust" is available for the identification of the mechanism for the classification between fake and benign user accounts



Figure2: Fake Profile Detection Models

The rank algorithm can be used for the analysis of the user's influence upon his friends. For the analysis of data from the various types of OSNs, big data can be introduced. By finding the challenges, the capability of algorithms is finding through the reduction of computational cost, time complexity, development of performance and enabling the local learning techniques. Recent experiments are also revealing the related difficulties with the "in-memory management" of big data (Ramalingam and Chinnaiyah, 2018). This is included with the "Hekton", "SAP HANA", "H-Store" and more. The "in-memory" data processing is also included with the big data analytics such as Spark, "Main Memory Map Reduce (M3R)" and real-time processing systems such as "Yahoo Simple Scalable Streaming System". In finding the challenges about data management, this can be found by finding the indexing possibilities, controlling concurrency, overflowing of data, and query processing.

According to Saatviket *et al.* 2020, the artificial neural network is mainly depending upon the three main factors, which are organizing the structure, institution and data component of a unit and data affiliation largeness. If one of these two parameters is right, then ANN lead can be described with the heap estimation. From the outset to self-assertive characteristics, the "heaps of net" are readied. The characteristics of a defined case's commitment can be easily determined by the data units (Khaled *et al.*, 2018). The yield of the net can be adjusted through a certain degree in a way through which the net is adjusted with a similar degree type. By bringing the yield estimation into the bet closer, the perfect yield's characteristics can be determined. In relating to this, the neutral structure of action can be introduced with consolidates with breaking down of the picture, "convolution neural structure", and a guide "neutral system. Therefore, the organizing map can be provided with the quantization of the picture test. Within the topological position, the input is closed to the head; spaces are shut in the yield space. The best mirror of improved acknowledgement and 3D face geometry is related to the "Spectral Regression Kernel Discriminate Analysis" (SRKDA). This SRKDA is also subjected to a loss of faith where the apparition of diagram assessment can be presented with the proposed method (Awasthi *et al.* 2020). The careful approaches cannot loosen up with the minimal and dimensional size issues, which will further improve the fuse extraction through the non-direct structures. Artificial intelligence is built on ANNs through which are disrupting the multiple traditional ways of doing things. By using learning algorithms, the ANN can incorporate different sectors.

2.2 SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

System : Pentium IV 2.4 GHz

Hard Disk : 40 GB.

- Floppy Drive : 1.44 Mb.
- Monitor : 15 VGA Colour.
- Mouse : Logitech.
- Ram : 512 Mb.

SOFTWARE REQUIREMENTS:

- **Operating System:** Windows
- **Coding Language:** Python 3.7

2.1 SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ◆ **ECONOMICAL FEASIBILITY**
- ◆ **TECHNICAL FEASIBILITY**
- ◆ **SOCIAL FEASIBILITY**

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.SYSTEM DESIGN

3.1 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

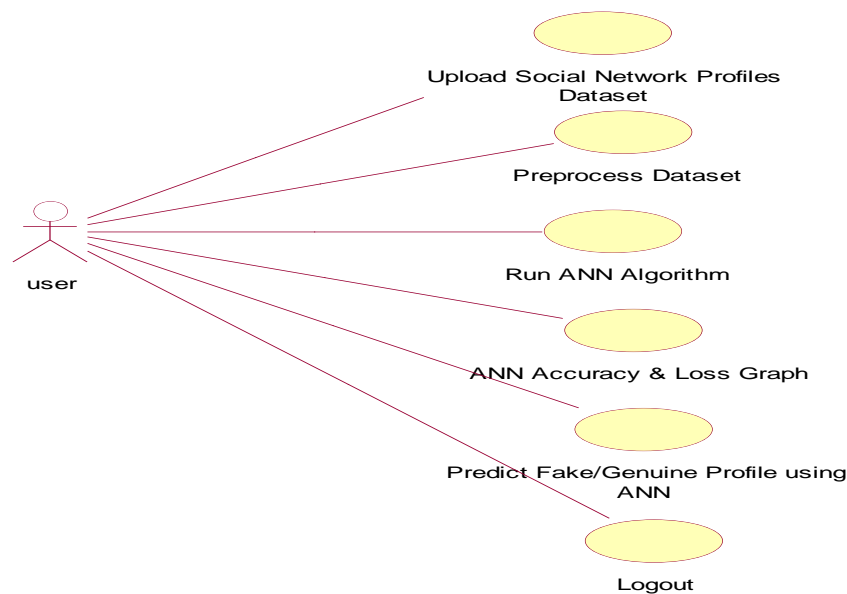
The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.

3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

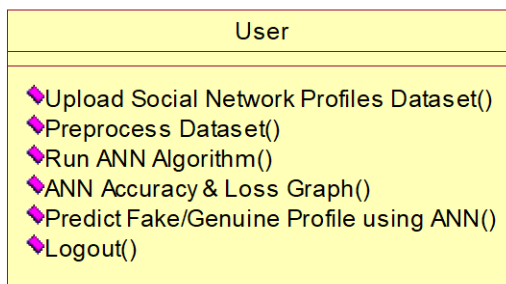
USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



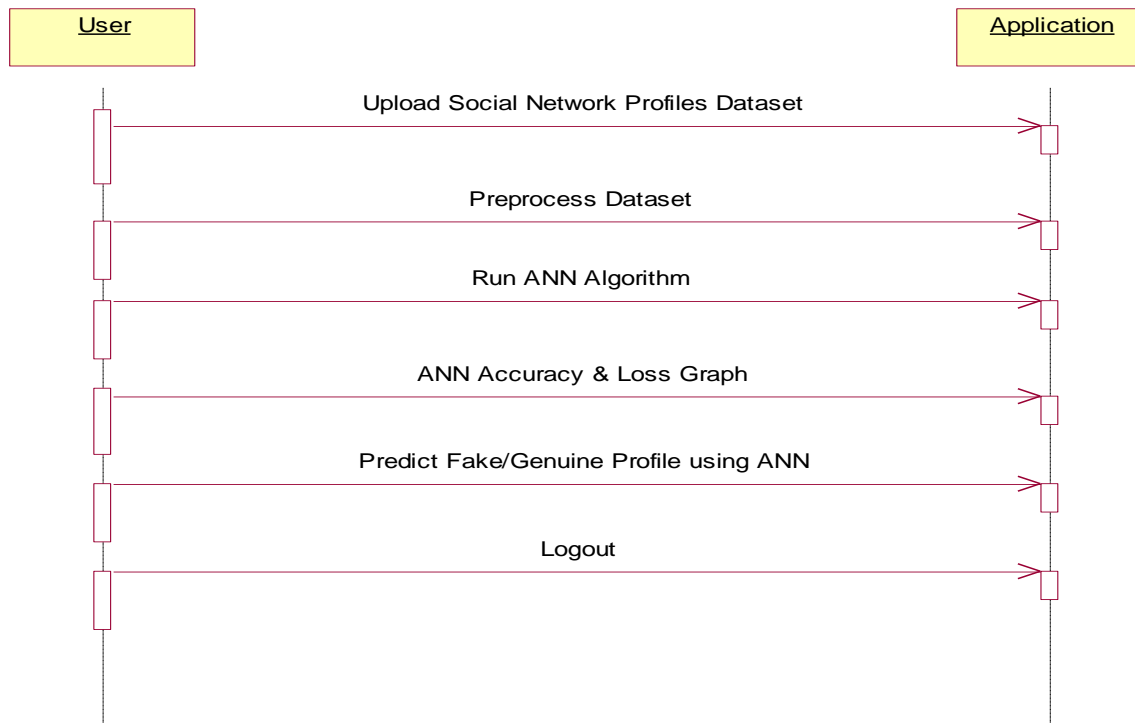
CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

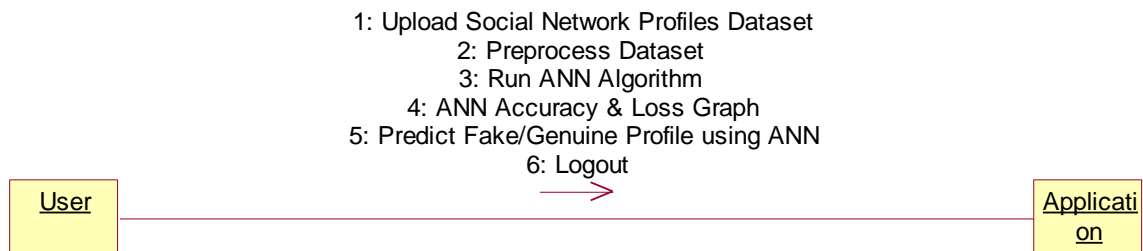


SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



Collaboration DIAGRAM:



3.3 IMPLEMENTATION:

MODULES:

Module Details:

1.Upload Social Network Profiles Dataset:

Using this module we will upload dataset to application

2.Preprocess Dataset:

Using this module, we will apply processing technique such as removing missing values and then split dataset into train and test where application use 80% dataset to train ANN and 20% dataset to test ANN prediction accuracy

3.Run ANN Algorithm:

Using this module, we will train ANN algorithm with train and test data and then train model will be generated and we can use this train model to predict fake accounts from new dataset.

4.ANN Accuracy & Loss Graph:

To train ANN model we are taking 200 epoch/iterations and then in graph we will plot accuracy/loss performance of ANN at each epoch/iteration.

5.Predict Fake/Genuine Profile using ANN: using this module we will upload new test data and then apply ANN train model to predict whether test data is genuine or fake.

5.SOFTWARE ENVIRONMENT

What is Python: -

Below are some facts about Python.

Python is currently the most widely used multi-purpose, high-level programming language.

Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.

Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.

Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber... etc.

The biggest strength of Python is huge collection of standard library which can be used for the following –

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like Opencv, Pillow)
- Web scraping (like Scrapy, BeautifulSoup, Selenium)
- Test frameworks
- Multimedia

Advantages of Python :-

Let's see how Python dominates over other languages.

1. Extensive Libraries

Python downloads with an extensive library and it contains code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don't have to write the complete code for that manually.

2. Extensible

As we have seen earlier, Python can be **extended to other languages**. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add **scripting capabilities** to our code in the other language.

4. Improved Productivity

The language's simplicity and extensive libraries render programmers **more productive** than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

5. IOT Opportunities

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet Of Things. This is a way to connect the language with the real world.

6. Simple and Easy

When working with Java, you may have to create a class to print **'Hello World'**. But in Python, just a print statement will do. It is also quite **easy to learn, understand, and code**.

This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.

7. Readable

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and **indentation is mandatory**. This further aids the readability of the code.

8. Object-Oriented

This language supports both the **procedural and object-oriented** programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the **encapsulation of data** and functions into one.

9. Free and Open-Source

Like we said earlier, Python is **freely available**. But not only can you **download Python** for free, but you can also download its source code, make changes to it, and even distribute it. It comes with an extensive collection of libraries to help you with your tasks.

10. Portable

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn't the same with Python. Here, you need to **code only once**, and you can run it anywhere. This is called **Write Once Run Anywhere (WORA)**. However, you need to be careful enough not to include any system-dependent features.

11. Interpreted

Lastly, we will say that it is an interpreted language. Since statements are executed one by one, **debugging is easier** than in compiled languages.

Any doubts till now in the advantages of Python? Mention in the comment section.

Advantages of Python Over Other Languages

1. Less Coding

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don't have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

2. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

The 2019 Github annual survey showed us that Python has overtaken Java in the most popular programming language category.

3. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and **machine learning**, automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

Disadvantages of Python

So far, we've seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let's now see the downsides of choosing Python over another language.

1. Speed Limitations

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in **slow execution**. This, however, isn't a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

2. Weak in Mobile Computing and Browsers

While it serves as an excellent server-side language, Python is much rarely seen on the **client-side**. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called **Carbannelle**.

The reason it is not so famous despite the existence of Brython is that it isn't that secure.

3. Design Restrictions

As you know, Python is **dynamically-typed**. This means that you don't need to declare the type of variable while writing the code. It uses **duck-typing**. But wait, what's that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can **raise run-time errors**.

4. Underdeveloped Database Access Layers

Compared to more widely used technologies like **JDBC (Java DataBase Connectivity)** and **ODBC (Open DataBase Connectivity)**, Python's database access layers are a bit underdeveloped. Consequently, it is less often applied in huge enterprises.

5. Simple

No, we're not kidding. Python's simplicity can indeed be a problem. Take my example. I don't do Java, I'm more of a Python person. To me, its syntax is so simple that the verbosity of Java code seems unnecessary.

This was all about the Advantages and Disadvantages of Python Programming Language.

History of Python : -

What do the alphabet and the programming language Python have in common? Right, both start with ABC. If we are talking about ABC in the Python context, it's clear that the programming language ABC is meant. ABC is a general-purpose programming language and programming environment, which had been developed in the Netherlands, Amsterdam, at the CWI (Centrum Wiskunde & Informatica). The greatest achievement of ABC was to influence the design of Python. Python was conceptualized in the late 1980s. Guido van Rossum worked that time in a project at the CWI, called Amoeba, a distributed operating system. In an interview with Bill Venners¹, Guido van Rossum said: "In the early 1980s, I worked as an implementer on a team building a language called ABC at Centrum voor Wiskunde en Informatica (CWI). I don't know how well people know ABC's influence on Python. I try to mention ABC's influence because I'm indebted to everything I learned during that project and to the people who worked on it." Later on in the same Interview, Guido van Rossum continued: "I remembered all my experience and some of my frustration with ABC. I decided to try to design a simple scripting language that possessed some of ABC's better properties, but without its problems. So I started typing. I created a simple virtual machine, a simple parser, and a simple runtime. I made my own version of the various ABC parts that I liked. I created a basic syntax, used indentation for statement grouping instead of curly braces or begin-end blocks, and developed a small number of powerful data types: a hash table (or dictionary, as we call it), a list, strings, and numbers."

What is Machine Learning : -

Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of *building models of data*.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models *tunable parameters* that can be

adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

Categories Of Machine Learning :-

At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into *classification* tasks and *regression* tasks: in classification, the labels are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as *clustering* and *dimensionality reduction*. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate and solve complex problems. On the other side, AI is still in its initial stage and haven't surpassed human intelligence in many aspects. Then the question

is that what is the need to make machine learn? The most suitable reason for doing this is, “to make decisions, based on data, with efficiency and scale”.

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programming logic, in the problems that cannot be programmed inherently. The fact is that we can't do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

Challenges in Machines Learning :-

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are –

Quality of data – Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task – Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons – As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems – Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of overfitting & underfitting – If the model is overfitting or underfitting, it cannot be represented well for the problem.

Curse of dimensionality – Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment – Complexity of the ML model makes it quite difficult to be deployed in real life.

Applications of Machines Learning :-

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML –

- Emotion analysis
- Sentiment analysis
- Error detection and prevention
- Weather forecasting and prediction
- Stock market analysis and forecasting
- Speech synthesis
- Speech recognition
- Customer segmentation
- Object recognition
- Fraud detection
- Fraud prevention
- Recommendation of products to customer in online shopping

How to Start Learning Machine Learning?

Arthur Samuel coined the term “**Machine Learning**” in 1959 and defined it as a “**Field of study that gives computers the capability to learn without being explicitly programmed**”.

And that was the beginning of Machine Learning! In modern times, Machine Learning is one of the most popular (if not the most!) career choices. According to Indeed, Machine Learning Engineer Is The Best Job of 2019 with a 344% growth and an average base salary of **\$146,085** per year.

But there is still a lot of doubt about what exactly is Machine Learning and how to start learning it? So this article deals with the Basics of Machine Learning and also the path you can follow to eventually become a full-fledged Machine Learning Engineer. Now let's get started!!!

How to start learning ML?

This is a rough roadmap you can follow on your way to becoming an insanely talented Machine Learning Engineer. Of course, you can always modify the steps according to your needs to reach your desired end-goal!

Step 1 – Understand the Prerequisites

In case you are a genius, you could start ML directly but normally, there are some prerequisites that you need to know which include Linear Algebra, Multivariate Calculus, Statistics, and Python. And if you don't know these, never fear! You don't need a Ph.D. degree in these topics to get started but you do need a basic understanding.

(a) Learn Linear Algebra and Multivariate Calculus

Both Linear Algebra and Multivariate Calculus are important in Machine Learning. However, the extent to which you need them depends on your role as a data scientist. If you are more focused on application heavy machine learning, then you will not be that heavily focused on

maths as there are many common libraries available. But if you want to focus on R&D in Machine Learning, then mastery of Linear Algebra and Multivariate Calculus is very important as you will have to implement many ML algorithms from scratch.

(b) Learn Statistics

Data plays a huge role in Machine Learning. In fact, around 80% of your time as an ML expert will be spent collecting and cleaning data. And statistics is a field that handles the collection, analysis, and presentation of data. So it is no surprise that you need to learn it!!! Some of the key concepts in statistics that are important are Statistical Significance, Probability Distributions, Hypothesis Testing, Regression, etc. Also, Bayesian Thinking is also a very important part of ML which deals with various concepts like Conditional Probability, Priors, and Posteriors, Maximum Likelihood, etc.

(c) Learn Python

Some people prefer to skip Linear Algebra, Multivariate Calculus and Statistics and learn them as they go along with trial and error. But the one thing that you absolutely cannot skip is Python! While there are other languages you can use for Machine Learning like R, Scala, etc. Python is currently the most popular language for ML. In fact, there are many Python libraries that are specifically useful for Artificial Intelligence and Machine Learning such as Keras, TensorFlow, Scikit-learn, etc.

So if you want to learn ML, it's best if you learn Python! You can do that using various online resources and courses such as **Fork Python** available Free on GeeksforGeeks.

Step 2 – Learn Various ML Concepts

Now that you are done with the prerequisites, you can move on to actually learning ML (Which is the fun part!!!) It's best to start with the basics and then move on to the more complicated stuff. Some of the basic concepts in ML are:

(a) Terminologies of Machine Learning

- **Model** – A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called a hypothesis.
- **Feature** – A feature is an individual measurable property of the data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc.
- **Target (Label)** – A target variable or label is the value to be predicted by our model. For the fruit example discussed in the feature section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.
- **Training** – The idea is to give a set of inputs(features) and it's expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.
- **Prediction** – Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label).

(b) Types of Machine Learning

- **Supervised Learning** – This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.
- **Unsupervised Learning** – This involves using unlabelled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.
- **Semi-supervised Learning** – This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.
- **Reinforcement Learning** – This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

Advantages of Machine learning :-

1. Easily identifies trends and patterns -

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Continuous Improvement

As **ML algorithms** gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data you have keeps growing, your algorithms learn to make more accurate predictions faster.

4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

Disadvantages of Machine Learning :-

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

4. High error-susceptibility

Machine Learning is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time. And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

Python Development Steps :-

Guido Van Rossum published the first version of Python code (version 0.9.0) at alt.sources in February 1991. This release included already exception handling, functions, and the core data types of list, dict, str and others. It was also object oriented and had a module system. Python version 1.0 was released in January 1994. The major new features included in this

release were the functional programming tools `lambda`, `map`, `filter` and `reduce`, which Guido Van Rossum never liked. Six and a half years later in October 2000, Python 2.0 was introduced. This release included list comprehensions, a full garbage collector and it was supporting unicode. Python flourished for another 8 years in the versions 2.x before the next major release as Python 3.0 (also known as "Python 3000" and "Py3K") was released. Python 3 is not backwards compatible with Python 2.x. The emphasis in Python 3 had been on the removal of duplicate programming constructs and modules, thus fulfilling or coming close to fulfilling the 13th law of the Zen of Python: "There should be one -- and preferably only one -- obvious way to do it." Some changes in Python 7.3:

- Print is now a function
- Views and iterators instead of lists
- The rules for ordering comparisons have been simplified. E.g. a heterogeneous list cannot be sorted, because all the elements of a list must be comparable to each other.
- There is only one integer type left, i.e. `int`. `long` is `int` as well.
- The division of two integers returns a float instead of an integer. `"/"` can be used to have the "old" behaviour.
- Text Vs. Data Instead Of Unicode Vs. 8-bit

Purpose :-

We demonstrated that our approach enables successful segmentation of intra-retinal layers—even with low-quality images containing speckle noise, low contrast, and different intensity ranges throughout—with the assistance of the ANIS feature.

Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

- Python is Interpreted – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive – you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

Modules Used in Project :-

Tensorflow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety of databases.

Pandas

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

Scikit – learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use.

Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

- Python is Interpreted – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive – you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

Install Python Step-by-Step in Windows and Mac :

Python a versatile programming language doesn't come pre-installed on your computer devices. Python was first released in the year 1991 and until today it is a very popular high-

level programming language. Its style philosophy emphasizes code readability with its notable use of great whitespace.

The object-oriented approach and language construct provided by Python enables programmers to write both clear and logical code for projects. This software does not come pre-packaged with Windows.

How to Install Python on Windows and Mac :

There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

Note: The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your **System Requirements**. Based on your system type i.e. operating system and based processor, you must download the python version. My system type is a **Windows 64-bit operating system**. So the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. [Download the Python Cheatsheet here.](#) The steps on how to install Python on Windows 10, 8 and 7 are **divided into 4 parts** to help understand better.

Download the Correct version into the system

Step 1: Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: <https://www.python.org>



Now, check for the latest and the correct version for your operating system.

Step 2: Click on the Download Tab.



Step 3: You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4

Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
Python 3.7.4	July 8, 2019	Download	Release Notes
Python 3.6.9	July 2, 2019	Download	Release Notes
Python 3.7.3	March 25, 2019	Download	Release Notes
Python 3.4.10	March 18, 2019	Download	Release Notes
Python 3.5.7	March 18, 2019	Download	Release Notes
Python 3.7.16	March 4, 2019	Download	Release Notes
Python 3.7.2	Dec. 24, 2018	Download	Release Notes

Step 4: Scroll down the page until you find the Files option.

Step 5: Here you see a different version of python along with the operating system.

Files

Version	Operating System	Description	MD5 Sum	File Size	GP6
Gzipped source tarball	Source release		68111671e5b3db4ae77b9ab01b0f09be	23017663	50G
XZ compressed source tarball	Source release		d33e4aae6097051c2eca45ee3604803	17131432	50G
macOS 64-bit/32-bit installer	Mac OS X	for Mac OS X 10.6 and later	6428b4fa7583daff1a442c8a8cee08e6	34898416	50G
macOS 64-bit installer	Mac OS X	for OS X 10.9 and later	5dd605c38217a457738f5e4a936b241f	28082845	50G
Windows help file	Windows		d63999573a2c56b2ac56cade6847c3d2	8131761	50G
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	9800c3cf6d9ec0b9abe83184a4072ba2	7504391	50G
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	a702b4b0ad76d4bdc3543a583e563400	26880348	50G
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	28c91c60ff6d72aeb53a3bd351b4bd2	1362904	50G
Windows x86 embeddable zip file	Windows		9fab3bd19841879fda9413574129d8	6741626	50G
Windows x86 executable installer	Windows		33cc802942a54446a3d8451476194789	25663848	50G
Windows x86 web-based installer	Windows		1b670cfa5d311d882c30983ea371d87c	1324608	50G

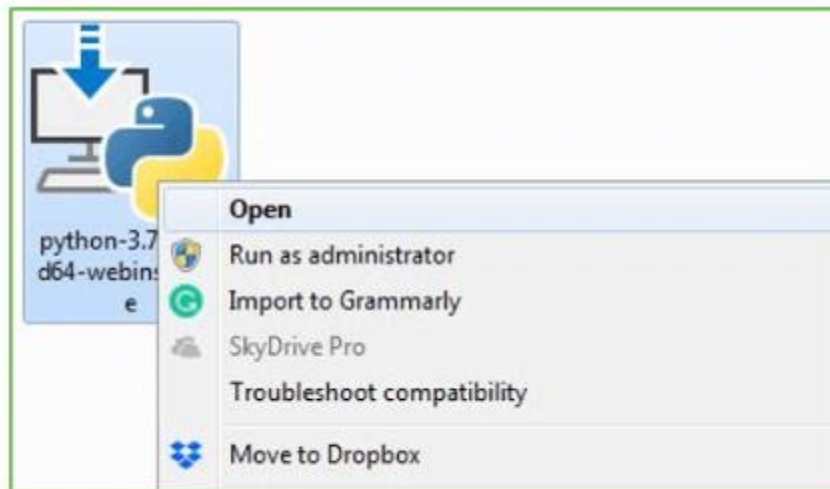
- To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.
- To download Windows 64-bit python, you can select any one from the three options: Windows x86-64 embeddable zip file, Windows x86-64 executable installer or Windows x86-64 web-based installer.

Here we will install Windows x86-64 web-based installer. Here your first part regarding which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e. Installation

Note: To know the changes or updates that are made in the version you can click on the Release Note Option.

Installation of Python

Step 1: Go to Download and Open the downloaded python version to carry out the installation process.



Step 2: Before you click on Install Now, Make sure to put a tick on Add Python 3.7 to PATH.



Step 3: Click on Install NOW After the installation is successful. Click on Close.



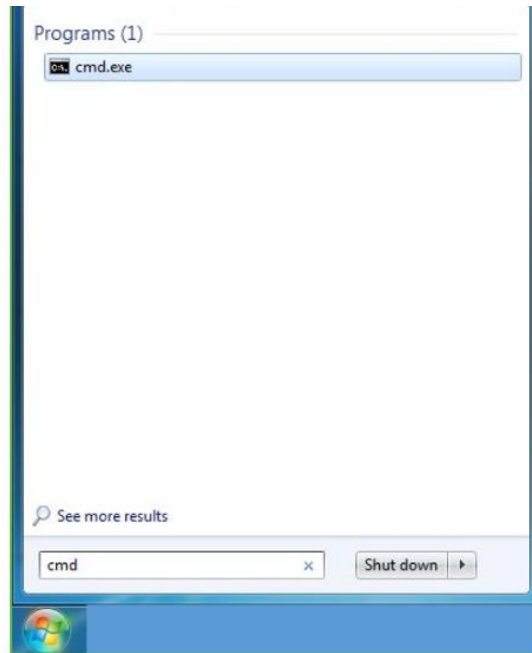
With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

Note: The installation process might take a couple of minutes.

Verify the Python Installation

Step 1: Click on Start

Step 2: In the Windows Run Command, type “cmd”.



Step 3: Open the Command prompt option.

Step 4: Let us test whether the python is correctly installed. Type **python -V** and press Enter.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\DELL>python -V
Python 3.7.4
C:\Users\DELL>_
```

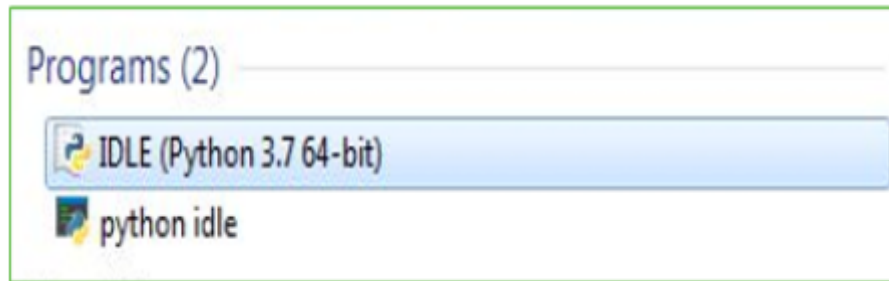
Step 5: You will get the answer as 3.7.4

Note: If you have any of the earlier versions of Python already installed. You must first uninstall the earlier version and then install the new one.

Check how the Python IDLE works

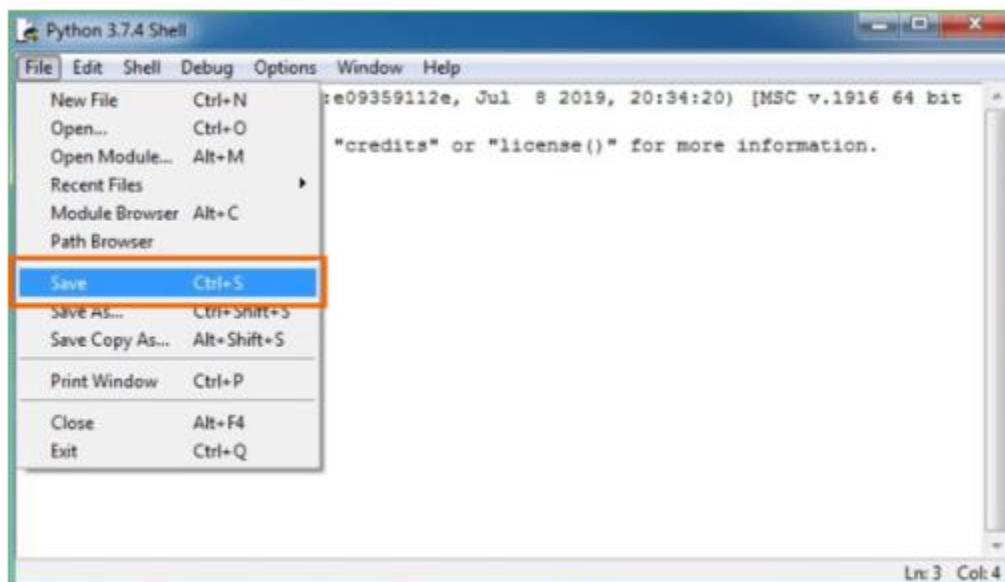
Step 1: Click on Start

Step 2: In the Windows Run command, type “python idle”.



Step 3: Click on IDLE (Python 3.7 64-bit) and launch the program

Step 4: To go ahead with working in IDLE you must first save the file. **Click on File > Click on Save**



Step 5: Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

Step 6: Now for e.g. **enter print**

6.SYSTEM TEST

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input: identified classes of valid input must be accepted.

Invalid Input: identified classes of invalid input must be rejected.

Functions: identified functions must be exercised.

Output: identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

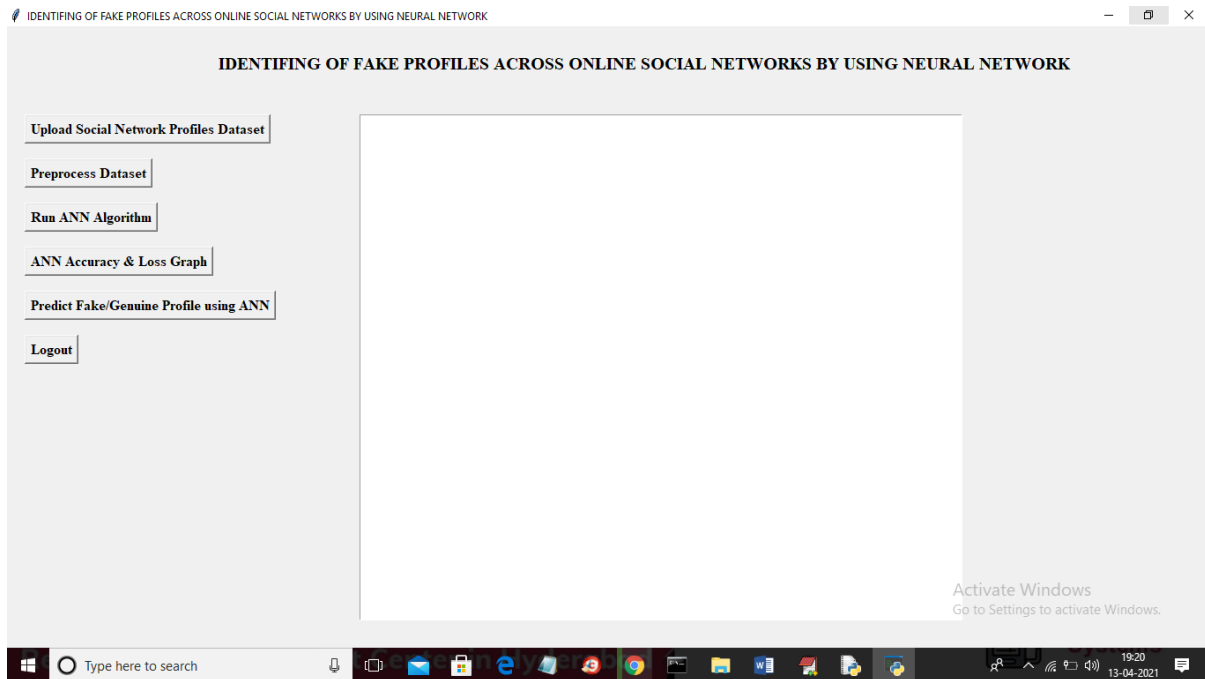
Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

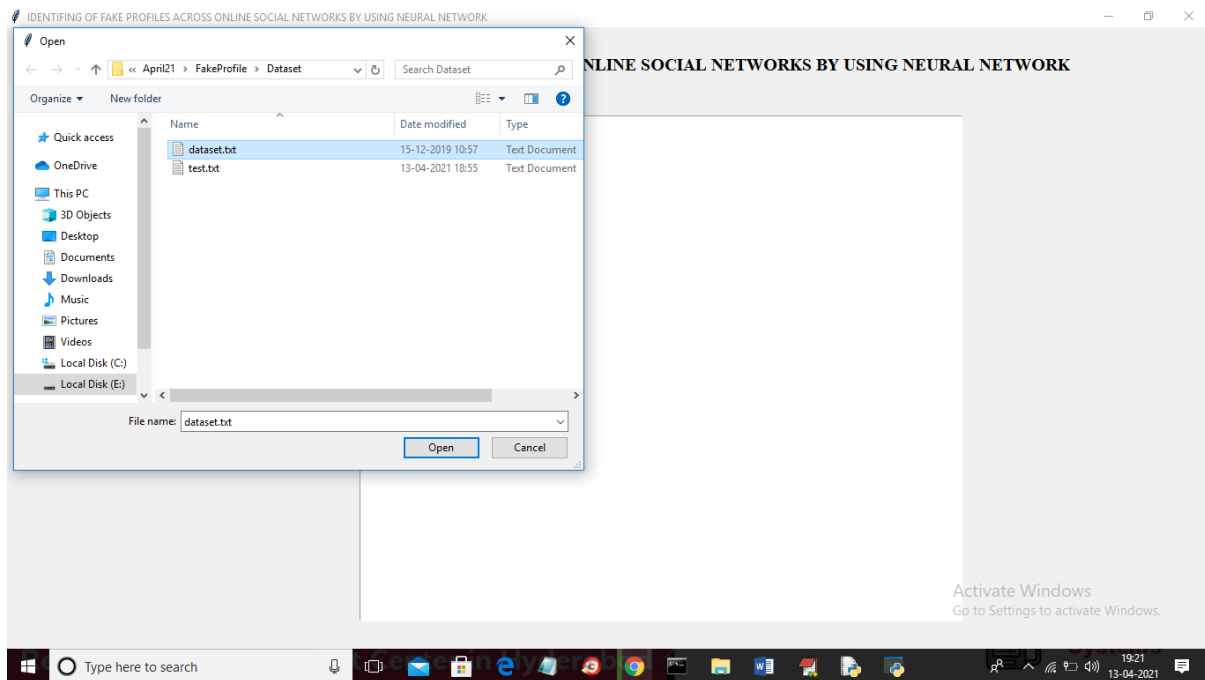
Test Results: All the test cases mentioned above passed successfully. No defects encountered.

7.SCREENSHOTS

To run project double click on 'run.bat' file to get below screen



In above screen click on 'Upload Social Network Profiles Dataset' button and upload dataset



In above screen selecting and uploading 'dataset.txt' file and then click on 'Open' button to load dataset and to get below screen

The screenshot shows a web application titled "IDENTIFYING OF FAKE PROFILES ACROSS ONLINE SOCIAL NETWORKS BY USING NEURAL NETWORK". On the left, there is a sidebar with buttons: "Upload Social Network Profiles Dataset", "Preprocess Dataset", "Run ANN Algorithm", "ANN Accuracy & Loss Graph", "Predict Fake/Genuine Profile using ANN", and "Logout". The main area displays a message "E:/venkat/2021/April21/FakeProfile/Dataset/dataset.txt loaded" above a table of dataset records. The table has columns: Account_Age, Gender, User_Age, Link_Desc, Status_Count, Friend_Count, Location, and Location_IP. The first five rows of data are shown. An "Activate Windows" watermark is visible in the bottom right corner.

	Account_Age	Gender	User_Age	Link_Desc	Status_Count	Friend_Count	Location	Location_IP
0	12	0	34	0	20370	2385	0	0 0
1	12	0	24	0	3131	381	0	0 0
2	12	0	59	0	4024	87	0	0 0
3	12	1	58	0	40586	622	0	0 0
4	12	0	59	0	2016	64	0	0 0

In above screen dataset loaded and displaying few records from dataset and now click on 'Preprocess Dataset' button to remove missing values and to split dataset into train and test part

The screenshot shows the same web application interface. The sidebar buttons remain the same. The main area now displays the following text: "Dataset contains total profiles : 600", "Total profiles used to train ANN algorithm : 480", and "Total profiles used to test ANN algorithm : 120". The "Activate Windows" watermark is still present in the bottom right corner.

In above screen we can see dataset contains total 600 records and application using 480 records for training and 120 records to test ANN and now dataset is ready and now click on 'Run ANN Algorithm' button to ANN algorithm

```
C:\Windows\system32\cmd.exe
Non-trainable params: 0

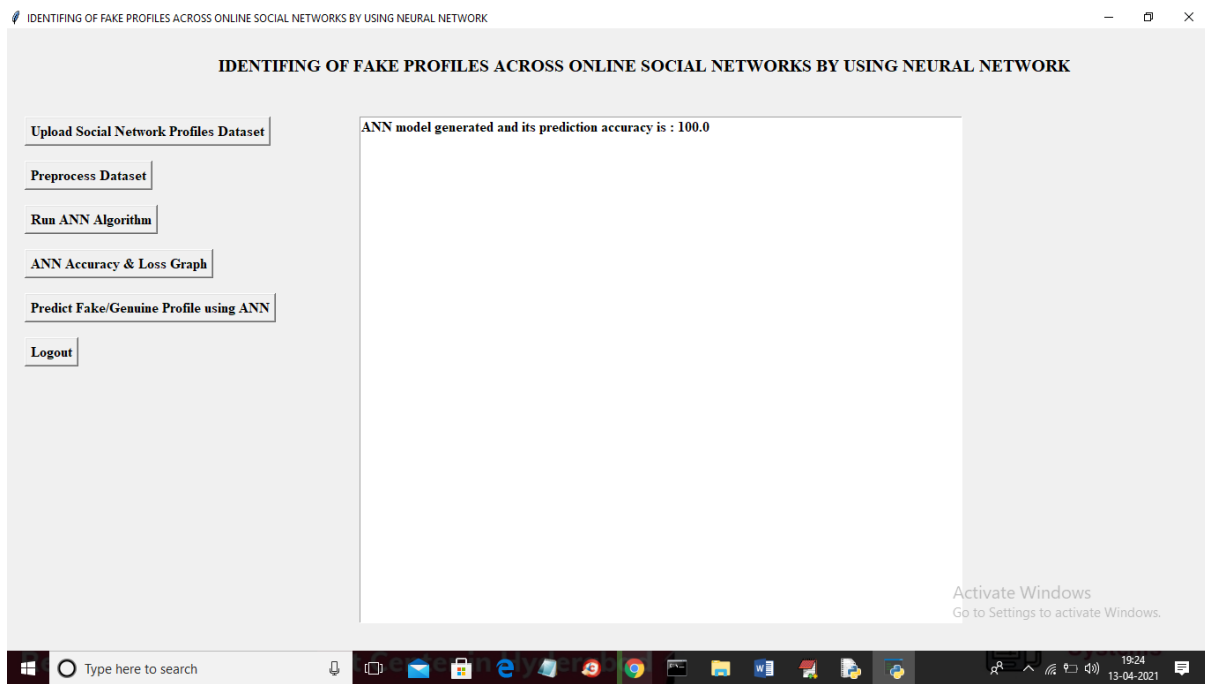
None
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.

Epoch 1/200
- 0s - loss: 10.2115 - accuracy: 0.9396
Epoch 2/200
- 0s - loss: 2.1024 - accuracy: 0.9375
Epoch 3/200
- 0s - loss: 8.0842 - accuracy: 0.9479
Epoch 4/200
- 0s - loss: 5.0597 - accuracy: 0.9604
Epoch 5/200
- 0s - loss: 2.8547 - accuracy: 0.9583
Epoch 6/200
- 0s - loss: 2.6966 - accuracy: 0.9417
Epoch 7/200
- 0s - loss: 9.7434 - accuracy: 0.9542
Epoch 8/200
- 0s - loss: 12.4945 - accuracy: 0.9604
Epoch 9/200
- 0s - loss: 5.1349 - accuracy: 0.9667
Epoch 10/200
- 0s - loss: 1.8743 - accuracy: 0.9521
Epoch 11/200
- 0s - loss: 2.7133 - accuracy: 0.9688
Epoch 12/200
- 0s - loss: 2.0212 - accuracy: 0.9479
Epoch 13/200
- 0s - loss: 1.3293 - accuracy: 0.9583
Epoch 14/200
- 0s - loss: 1.3460 - accuracy: 0.9688
Epoch 15/200
- 0s - loss: 1.0952 - accuracy: 0.9375
Epoch 16/200
- 0s - loss: 1.7365 - accuracy: 0.9604
Epoch 17/200
- 0s - loss: 1.3218 - accuracy: 0.9396
Epoch 18/200
- 0s - loss: 1.2545 - accuracy: 0.9688
Epoch 19/200
```

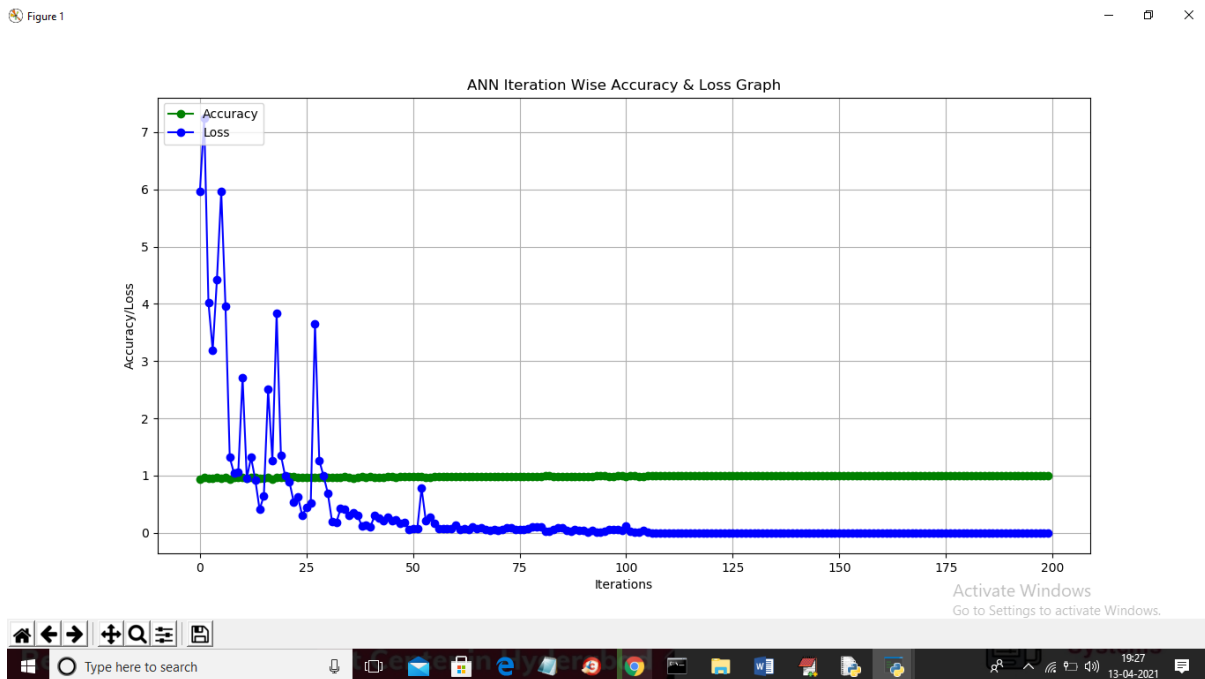
In above screen we can see ANN start iterating model generation and at each increasing epoch we can see accuracy is getting increase and loss getting decrease.

```
C:\Windows\system32\cmd.exe
- 0s - loss: 9.8744e-07 - accuracy: 1.0000
Epoch 187/200
- 0s - loss: 9.3032e-07 - accuracy: 1.0000
Epoch 188/200
- 0s - loss: 8.4067e-07 - accuracy: 1.0000
Epoch 189/200
- 0s - loss: 8.1806e-07 - accuracy: 1.0000
Epoch 190/200
- 0s - loss: 7.5871e-07 - accuracy: 1.0000
Epoch 191/200
- 0s - loss: 6.9066e-07 - accuracy: 1.0000
Epoch 192/200
- 0s - loss: 6.4373e-07 - accuracy: 1.0000
Epoch 193/200
- 0s - loss: 6.0225e-07 - accuracy: 1.0000
Epoch 194/200
- 0s - loss: 5.6972e-07 - accuracy: 1.0000
Epoch 195/200
- 0s - loss: 5.1980e-07 - accuracy: 1.0000
Epoch 196/200
- 0s - loss: 5.1309e-07 - accuracy: 1.0000
Epoch 197/200
- 0s - loss: 5.4190e-07 - accuracy: 1.0000
Epoch 198/200
- 0s - loss: 3.9562e-07 - accuracy: 1.0000
Epoch 199/200
- 0s - loss: 4.1127e-07 - accuracy: 1.0000
Epoch 200/200
- 0s - loss: 3.8047e-07 - accuracy: 1.0000
120/120 [=====] - 0s 609us/step
100.0
```

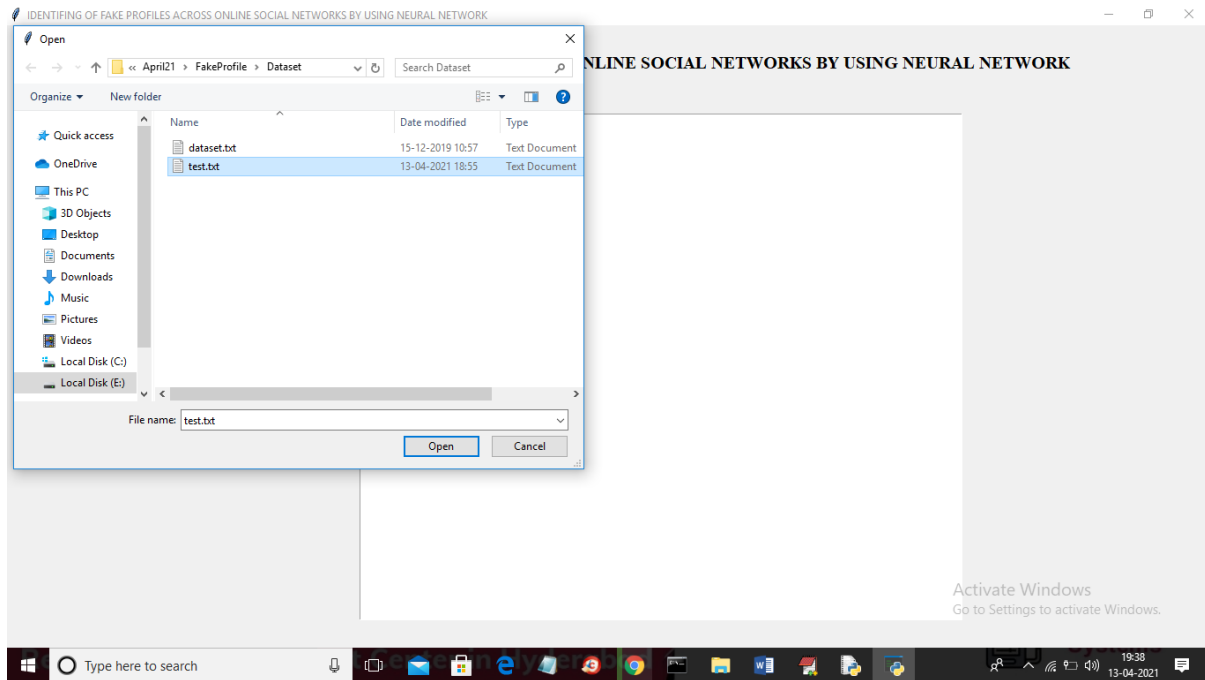
In above screen we can see after 200 epoch ANN got 100% accuracy and in below screen we can see final ANN accuracy



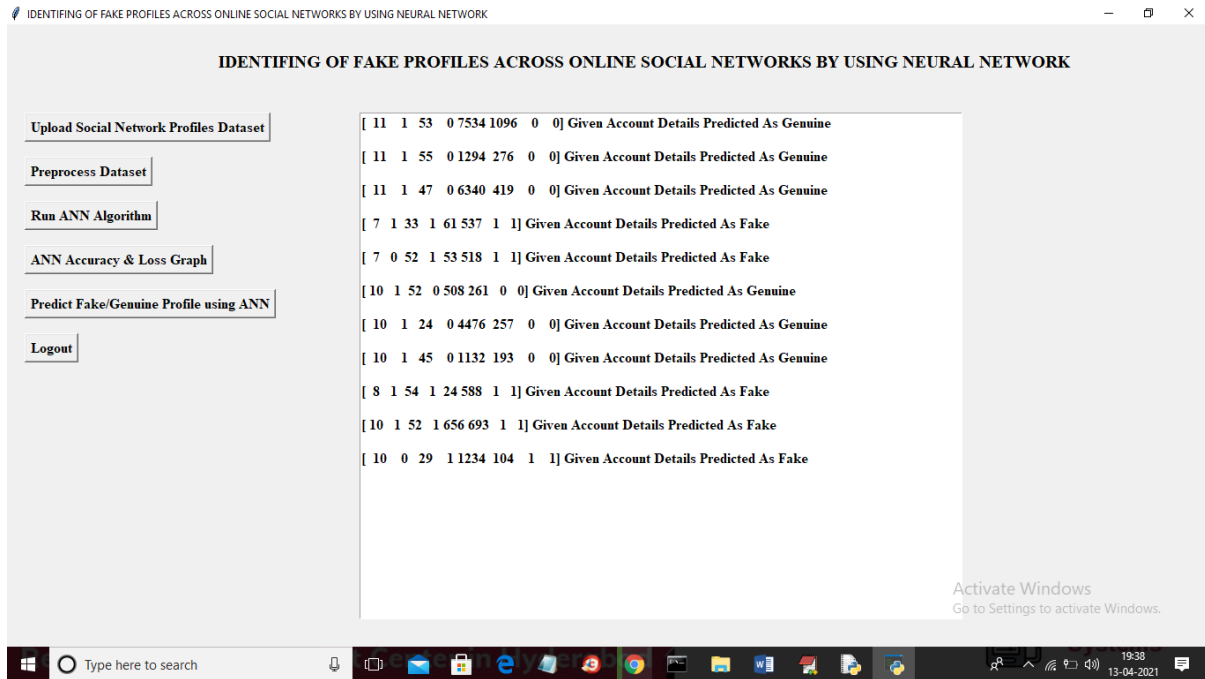
In above screen ANN model generated and now click on 'ANN Accuracy & Loss Graph' button to get below graph



In above graph x-axis represents epoch and y-axis represents accuracy/loss value and in above graph green line represents accuracy and blue line represents loss value and we can see accuracy was increase from 0.90 to 1 and loss value decrease from 7 to 0.1. Now model is ready and now click on 'Predict Fake/Genuine Profile using ANN' button to upload test data and then ANN will predict below result



In above screen we are selecting and uploading 'test.txt' file and then click on 'Open' button to load test data and to get below prediction result



In above screen in square bracket we can see uploaded test data and after square bracket we can see ANN prediction result as genuine or fake

Significance of Research

ANNs are basically known as lone performers, which are not intended in the production of the general network types. This software is used for practical application through its networks. The primary focus is on forecasting and data mining. The software tools are used as -

1. Darknet,
2. NeuroSolutions,
3. Neural Designer,
4. Keras,
5. Neuroph,
6. Tflearn,
7. Torch,
8. "Stuttgart Neural Network Simulator",
9. ConvNetJS,
10. NVIDIA DIGITS,

The ANN process has the ability in the relearning process according to the newer data types. Due to the uncertainty and complexity, it is difficult in defining a particular analytical model. In the elaborate ritual, a powerful computer-based application can be used. Therefore, the optimization technique's principle lies in the optimization process through which both the constraints and object functions are evaluated into the simulation model (Wanda and Jie, 2020). For the combined simulation, the optimization techniques and ANN need to be provided with the practical means for higher complex optimization. In searching for the solution space, the 'multi-objective optimization algorithm" or NSGA-II is used with adaptive local search. In discrediting the event simulation model, both input-output data is used for the generation of ANN in approximating the object function. Acting as an intelligent brain, this can train simulated data and accurate models.

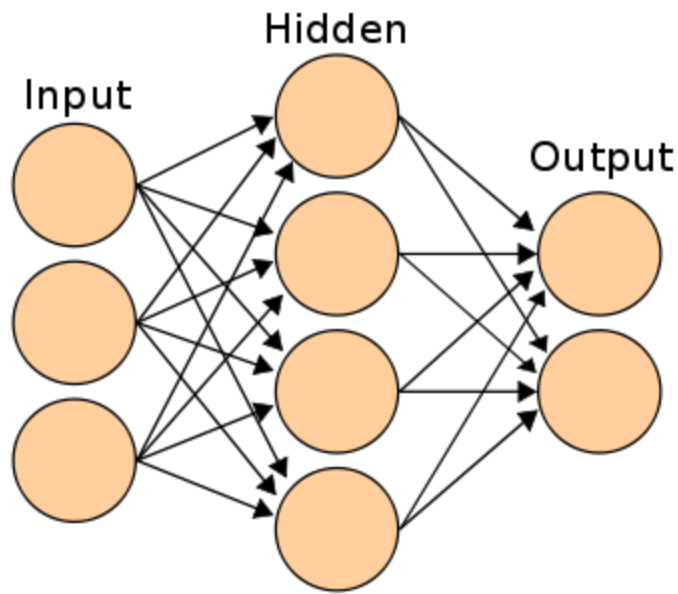


Figure3: ANN Framework

(Source:<https://www.analyticsvidhya.com/wp-content/uploads/2014/10/ANN.png>)

In between the nodes, the linkages are considered as the main factors (Zhang *et al.* 2020). By finding random weights of the linkages at the start of the algorithm, using the inputs for finding the linkages, searching the errors at the output nodes, weight calibration in between hidden and input nodes, defining the final linkage weights for the scoring of activation rate the framework is structured. Apart from this, by using hidden nodes and their linkages with the output, the output nodes' activation nodes can be found out.

4: Required Resources

In finding the resources, there are multiple modules that can be used. As the social network is a general site, by implementing artificial neural networks, different kinds of modules can be used in the detection of fake profiles. PyBrain is known as a modular within the machine learning library in using Python. Comparing the algorithm with predefined environments can offer better machine learning tasks. Scikit-learn are used for machine learning through Python (Meligy *et al.*, 2017). In predictive data analysis, it is considered as efficient tools. The sexmachine was created for publishing Python 3 compatible versions into PyPi. Without bugging, it can add definite improvements. In relation to this matplotlib is considered a comprehensive library used for animated, static, and interactive visualizations in python. This can make easy and more challenging things more efficiently to create. The ipython notebook is also known as the Jupiter notebook. In the computational environment, it can be combined with the execution of codes, plots and mathematics. Therefore ipython is also known as an interactive shell of python. A Jupiter kernel works with the code in the notebook.

5: Required Skills

The activity of this related technique is from translating web pages into three virtual assistants to order groceries while conversing with chatbots in solving problems. Email servers are also using ANNs and deleting spam from the user inbox. Chatbots are also developed with ANNs as a "natural language processing". Pandas in the package of python are delivering their flexible, fast and flexible data structure in working with the level data types. For working with the array types, NumPy is used in the python library. The working function lies in the domain of "linear algebra". In relation to this pip is known as the package manager of python. This is used as a distributed

part of the standard library (Kaur and Sabharwal, 2018). Apart from this, the knowledge is required about Java, Python more clearly. In using the modules and packages, depth knowledge and preferred system configuration are preferable. Python 3.9 is used as the best version. Therefore, the ram, hard disk capacity, and the IDE packages are necessary for working with the programs.

6: Project Plan

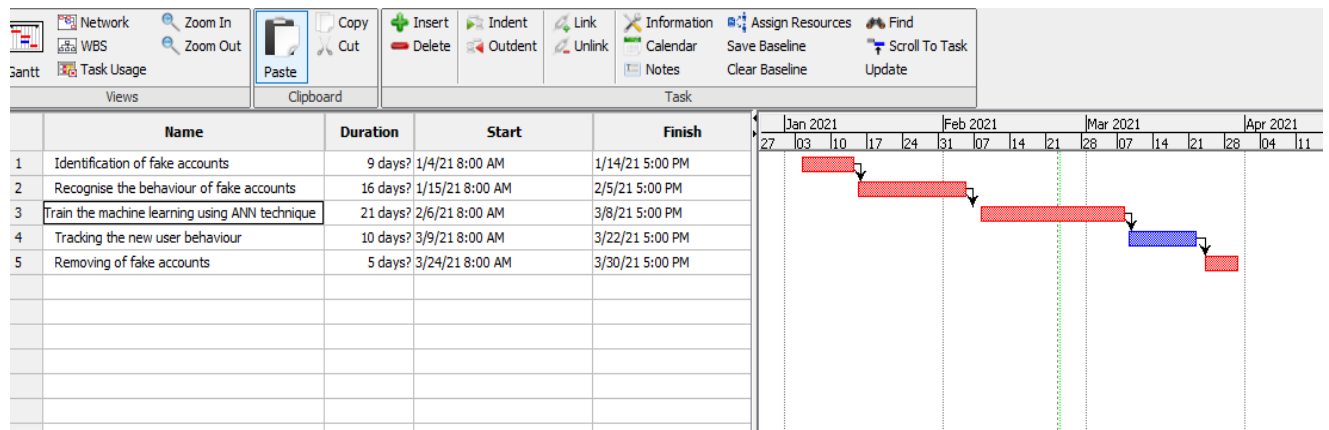


Figure4: Gantt chart

(Source: Project Libra)

Reference list

Journals

Awasthi, S., Shanmugam, R., Jena, S.R. and Srivastava, A., 2020. Review of Techniques to Prevent Fake Accounts on Social Media.

Hajdu, G., Minoso, Y., Lopez, R., Acosta, M. and Elleithy, A., 2019, May. Use of Artificial Neural Networks to Identify Fake Profiles. In *2019 IEEE Long Island Systems, Applications and Technology Conference (LISAT)* (pp. 1-4). IEEE.

Kaur, J. and Sabharwal, M., 2018. Spam detection in online social networks using feed forward neural network. In *RSRI conference on recent trends in science and engineering* (Vol. 2, pp. 69-78).

Khaled, S., El-Tazi, N. and Mokhtar, H.M., 2018, December. Detecting fake accounts on social media. In *2018 IEEE International Conference on Big Data (Big Data)* (pp. 3672-3681). IEEE.

Meligy, A.M., Ibrahim, H.M. and Torky, M.F., 2017. Identity verification mechanism for detecting fake profiles in online social networks. *Int. J. Comput. Netw. Inf. Secur.(IJCNIS)*, 9(1), pp.31-39.

Ramalingam, D. and Chinnaiah, V., 2018. Fake profile detection techniques in large-scale online social networks: A comprehensive review. *Computers & Electrical Engineering*, 65, pp.165-177.

Wanda, P. and Jie, H.J., 2020. DeepProfile: Finding fake profile in online social network using dynamic CNN. *Journal of Information Security and Applications*, 52, p.102465.

Zhang, J., Dong, B. and Philip, S.Y., 2020, April. Fakedetector: Effective fake news detection with deep diffusive neural network. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)* (pp. 1826-1829). IEEE.